



实验报告

学院：计算机科学技术专业：计算机应用技术 2025 年 4 月 14 日

姓名	高梦珊	学号	2024040731222	
班级	一班	指导老师	孔德华老师	
课程名称	C 语言程序设计实践		成绩	
实验名称	函数-1			

1. 实验目的

- 理解使用函数的意义；
- 掌握函数的定义方法；
- 掌握函数调用中的参数传递规则。

2. 实验内容

- 编写一个程序，通过函数调用的方式计算 $y=|x|$ 。
- 在主函数中输入一个整数，在子函数中判定该数据的奇偶性后，将判定结果在主函数中输出。
- 编写一个函数，计算任一输入的整数的各位数字之和。其中，主函数包括输入、输出和调用该函数。
- 在主函数中输入两个实数，在该程序中用四个子函数实现求这两个数据的和、差、积、商，并在主函数中将计算结果输出。
- 用函数实现该功能：判断某年是否为闰年，输入输出信息均在主函数中实现。
- 编写一个函数，判断某一个四位数是不是玫瑰花数（所谓玫瑰花数即该四位数各位数字的四次方和恰好等于该数本身，如： $1634=1^4 + 6^4 + 3^4 + 4^4$ ）。在主函数中从键盘任意输入一个四位数，调用该函数，判断该数是否为玫瑰花数，若是则输出“yes”，否则输出“no”。
- 用递归法计算菲波那契数列（Fibonacci 序列）的前 20 项。

$$\begin{aligned}f(1) &= 1 & (n=1) \\f(2) &= 1 & (n=2) \\f(n) &= f(n-1) + f(n-2) & (n \geq 3)\end{aligned}$$

- 利用函数调用求两整数的最大公约数和最小公倍数。

3. 实验环境

① windows 10

② VC++6.0

4. 实验方法和步骤（含设计）

1、编写一个程序，通过函数调用的方式计算 $y=|x|$ 。

The screenshot shows the Dev-C++ IDE interface. The code editor displays a C program named '1寒树abs.c' which calculates the absolute value of a float input. The terminal window shows the program's output for an input of -5, displaying the result as 5.000000.

```
#include<stdio.h>
float abs(float);
int main()
{
    float x;
    printf("请输入待计算的数: \n");
    scanf("%f",&x);
    printf("绝对值为: %f",abs(x));
}

float abs(float x)
{
    if(x>=0)
        return x;
    else
        return -1*x;
}
```

2、在主函数中输入一个整数，在子函数中判定该数据的奇偶性后，将判定结果在主函数中输出。

The screenshot shows the Dev-C++ IDE interface. The code editor displays a C program named '2寒树奇偶数.c' which uses a main function to call an odd/even checker function. The terminal window shows the program's output for an input of 9, identifying it as an odd number.

```
#include<stdio.h>
void oe(int);
int main()
{
    int n;
    printf("请输入待检验的数: \n");
    scanf("%d",&n);
    printf("该数");
    oe(n);
    printf("奇数");
}

void oe(int n)
{
    if(n%2==1)
        printf("是");
    else
        printf("不是");
}
```

3、编写一个函数，计算任一输入的整数的各位数字之和。其中，主函数包括输入、输出和调用该函数。

The screenshot shows the Dev-C++ IDE interface. The code in the editor is:

```
1 #include<stdio.h>
2 int cardinality_sum(int);
3 int main()
4 {
5     int n;
6     printf("请输入待计算的数: \n");
7     scanf("%d",&n);
8     printf("该数各位数字之和为: %d",cardinality_sum(n));
9 }
10
11 int cardinality_sum(int n)
12 {
13     int sum=0;
14     for(;n!=0;)
15     {
16         sum+=n%10;
17         n/=10;
18     }
19     return sum;
20 }
```

The terminal window shows the output of the program:

```
请输入待计算的数:
16329
该数各位数字之和为: 22
Process exited after 15 seconds with return value 22
请按任意键继续... |
```

4、在主函数中输入两个实数，在该程序中用四个子函数实现求这两个数据的和、差、积、商，并在主函数中将计算结果输出

The screenshot shows the Dev-C++ IDE interface. The code in the editor is:

```
1 #include<stdio.h>
2 float ASMD(float,char,float);
3 int main()
4 {
5     float x,y;
6     char c;
7     printf("请按格式输入待计算的数: x ? y \n");
8     scanf("%f%c%f",&x,&c,&y);
9     printf("结果为: %.2f%c%.2f=%#.2f",x,c,y,ASMD(x,c,y));
10 }
11
12 float ASMD(float x,char c,float y)
13 {
14     float f=0;
15     if(c=='+')
16         return x+y;
17     else if(c=='-')
18         return x-y;
19     else if(c=='*')
20         return x*y;
21     else if(c=='/'&&y!=0)
22         return x/y;
23     else
24     {
25         printf("输入错误请重新输入 \n");
26         return 0;
27     }
28 }
```

The terminal window shows the output of the program:

```
请输入待计算的数: x ? y
9/6
结果为: 9.00/6.00=1.50
Process exited after 11.49 seconds with return value 22
请按任意键继续... |
```

```
请按格式输入待计算的数: x ? y
9/0
输入错误请重新输入
结果为: 9.00/0.00=0.00
```

5、用函数实现该功能：判断某年是否为闰年，输入输出信息均在主函数中实现。

The screenshot shows the Dev-C++ IDE interface. The code in the editor is:

```
1 #include<stdio.h>
2 void leap_year(int);
3 int main()
4 {
5     int n;
6     printf("请输入待判断的年份: \n");
7     scanf("%d",&n);
8     leap_year(n);
9     return 0;
10 }
11
12 void leap_year(int year)
13 {
14     if(year%4==0&&year%100!=0||year%400==0)
15         printf("该年是闰年\n");
16     else
17         printf("该年不是闰年\n");
18 }
```

The terminal window shows the output of running the program with the input 2924, which is a leap year.

```
E:\Code\Karigar\25C\小作业\4\5寒树润.c
请输入待判断的年份:
2924
该年是闰年
```

The status bar at the bottom indicates the process exited after 3.656 seconds.

6、编写一个函数，判断某一个四位数是不是玫瑰花数

The screenshot shows the Dev-C++ IDE interface. The code in the editor is:

```
1 #include<stdio.h>
2 void Rose(int);
3 int Exponent4(int);
4 int main()
5 {
6     int n;
7     printf("请输入待判断的四位数: \n");
8     scanf("%d",&n);
9     Rose(n);
10    return 0;
11 }
12
13 void Rose(int n)
14 {
15     int i,m=n,sum=0;
16     for(i=0;i<4;i++)
17     {
18         sum+=Exponent4(n%10);
19         n/=10;
20     }
21     if(m==sum)
22         printf("yes\n");
23     else
24         printf("no\n");
25 }
26
27 int Exponent4(int n)
28 {
29     return n*n*n*n;
30 }
```

The terminal window shows the output of running the program with the input 1634, which is a rose number.

```
E:\Code\Karigar\25C\小作业\4\6寒树玫瑰花数.c
请输入待判断的四位数:
1634
yes
```

The status bar at the bottom indicates the process exited after 5.055 seconds.

7、用递归法计算斐波那契数列（Fibonacci 序列）的前 20 项。

The screenshot shows the Dev-C++ IDE interface. The code in the editor is:

```
#include<stdio.h>
int Fibonacci(int);
int main()
{
    int i,Ent=1;
    printf("斐波那契前二十项为: \n");
    for(i=1;i<=20;i++)
    {
        printf("%d\t",Fibonacci(i));
        if(Ent%5==0)
            printf("\n");
        Ent++;
    }
    return 0;
}
int Fibonacci(int n)
{
    if(n==1||n==2)
        return 1;
    else
        return Fibonacci(n-1)+Fibonacci(n-2);
}
```

The output window displays the first 20 terms of the Fibonacci sequence:

1	1	2	3	5
8	13	21	34	55
89	144	233	377	610
987	1597	2584	4181	6765

Below the output window, the message "Process exited after 5.052 seconds with return value 0" is shown.

The bottom status bar indicates the code has been fully analyzed.

8、利用函数调用求两整数的最大公约数和最小公倍数。

The screenshot shows the Dev-C++ IDE interface. The code in the editor is:

```
#include<stdio.h>
int GCD(int,int);
int LCM(int,int);
int main()
{
    int m1,m2;
    printf("请输入计算的两数: \n");
    scanf("%d%d",&m1,&m2);
    printf("最大公约数与最小公倍数分别为 : \n");
    printf("%d , %d ",GCD(m1,m2),LCM(m1,m2));
    return 0;
}
int GCD(int m1,int m2)
{
    int r;
    if(m2>m1){r=m2;m2=m1;m1=r;}
    for(;r!=0;)
    {
        r=m1%m2;
        m1=m2;
        m2=r;
    }
    return m1;
}
int LCM(int m1,int m2)
{
    return m1*m2/GCD(m1,m2);
}
```

The output window shows the input "23 11" and the output "最大公约数与最小公倍数分别为 : 1 , 253".

Below the output window, the message "Process exited after 9.505 seconds with return value 0" is shown.

The bottom status bar indicates the code has been fully analyzed.

5. 程序及测试结果

6. 实验分析与体会

本次编程实践使我全面认识到函数在程序设计中的核心地位，并深刻理解了模块化编程的实际价值。通过完成不同功能的函数设计与调用，不仅提升了代码组织能力，更培养了将复杂问题拆解为独立功能单元的思维模式。

在实践过程中，我逐步掌握了参数传递与返回值处理的技巧。例如，在构建四则运算体系时，将加减乘除运算封装为独立函数，并通过字符参数驱动运算选择，这种设计方式使主程序逻辑简洁高效。数值处理方面，通过取模运算与循环结构的配合，实现了数字分解与特征提取，这在计算数位之和与四位数验证等任务中得到充分应用。特别是玫瑰花数的判断程序，通过创建指数计算函数与数值分解函数的协作，体现了代码复用的优越性。

递归算法的实现过程带来了重要启示。斐波那契数列的递归生成虽然代码简洁，但实际调试时的堆栈观察使我对递归机制有了直观认识。而最大公约数的辗转相除法实现，则展示了函数嵌套调用的灵活性——通过将公约数计算封装为独立函数，公倍数的求解只需进行简单公式运算，这种函数间的协同显著提升了开发效率。

逻辑构建能力的提升尤为明显。闰年判断程序看似简单，实则蕴含多条件组合的严谨性：必须同时考虑4的倍数、100的非整除特例与400的整除规则，这要求开发者具备精确的逻辑表达能力。数据验证机制的建立也得到强化，如四则运算中对除数为零的预判处理，体现了程序的健壮性设计理念。

通过调试各类函数模块，我建立了系统化的验证思维。每个功能单元独立测试后，再逐步集成到主程序中进行联合调试，这种分阶段验证方法大幅降低了错误排查难度。例如在绝对值函数中，通过正负数多组测试数据的验证，确保了边界条件的正确处理。

本次实验使我深刻体会到，函数不仅是代码复用的载体，更是构建程序逻辑的基石。从基础数值处理到复杂算法实现，合理的函数划分能显著提升开发效率与代码可维护性。这些经验为后续学习面向对象编程奠定了重要基础，也让我对“分而治之”的编程哲学有了更深刻的理解。

实验日期：2025年4月14日

教师评语

签名：

年 月 日